

Algoritma Natural Language Processing Untuk Aplikasi Penerjemah (Indonesia – Jawa) Menggunakan Metode Speech Processing

Aria Andros Ghirrid ^{1*}, Ratih Titi Komala Sari ², Rima Tamara Aldisa ³

^{1,2,3} Program Studi Informatika, Fakultas Teknologi Komunikasi dan Informatika, Universitas Nasional, Kota Jakarta Selatan, Daerah Khusus Ibukota Jakarta, Indonesia.

article info

Article history:

Received 11 February 2024

Received in revised form

14 March 2024

Accepted 1 May 2024

Available online July 2024.

DOI:

<https://doi.org/10.35870/jti.k.v8i3.2244>.

Keywords:

Javanese language; Speech to Text translation; Natural Language Processing (NLP).

Kata Kunci:

Bahasa Jawa; Speech to Text translation; Natural Language Processing (NLP).

abstract

Javanese is one of the most widely used regional languages in Indonesia. Apart from that, in several countries such as Suriname, the Netherlands and Malaysia, Javanese is used. The aim of this research is to create a translator application, allowing users to convert speech into written text. In creating a Speech to Text translation application, the Natural Language Processing (NLP) algorithm has a major role. NLP algorithms are used to understand the structure of human language, recognize entities, and translate complex speech into readable text. This study discusses the design and development of a Speech to Text translator application using the NLP algorithm. This application uses deep learning models that have been trained to understand user language variations, accents and intonation.

abstrak

Bahasa jawa merupakan salah satu bahasa daerah yang paling banyak digunakan di indonesia. Selain itu di daerah beberapa negara seperti Suriname, Belanda, dan Malaysia menggunakan bahasa jawa. Tujuan penelitian ini adalah membuat aplikasi penerjemah, memungkinkan pengguna untuk mengubah ucapan menjadi teks tertulis. Dalam pembuatan aplikasi penerjemah Speech to Text, Algoritma Natural Language Processing (NLP) memiliki peran utama. Algoritma NLP digunakan untuk memahami struktur bahasa manusia, mengenali entitas, dan menerjemahkan ucapan kompleks menjadi teks yang dapat dibaca. Studi ini membahas desain dan pengembangan aplikasi penerjemah Speech to Text dengan menggunakan algoritma NLP. Aplikasi ini menggunakan model-model deep learning yang telah terlatih untuk memahami variasi bahasa, aksen, dan intonasi pengguna.

Corresponding Author. Email: ariaandross68@gmail.com ^{1}.

© E-ISSN: 2580-1643.

Copyright @ 2024 by the authors of this article. Published by Lembaga Otonom Lembaga Informasi dan Riset Indonesia (KITI INFO dan Riset). This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License. 

1. Latar Belakang

Pada kehidupan yang berteknologi maju ini, bahasa Jawa hampir hilang di Generasi milenial yang berasal dari keluarga asli jawa yang merantau ke Jakarta, mempelajari bahasa jawa ini kurang begitu diminati dan seringkali dianggap remeh. Di kalangan generasi milenial lebih memilih mempelajari bahasa asing, dari pada bahasa daerah. Kurangnya ketertarikan dalam menguasai bahasa daerah, baik budaya ataupun bahasa yang dimana saat teknologi belum berkembang pesat seperti sekarang ini, masyarakat bergantung pada media dalam mempelajari bahasa daerah seperti melalui, majalah, tabloid, radio dan televisi. Dengan berbagai permasalahan tersebut aplikasi penerjemah Indonesia-Jawa menjadi sebuah alat yang berharga dalam memfasilitasi komunikasi antara orang yang berbicara bahasa Indonesia dan bahasa Jawa, serta mendukung pelestarian bahasa dan budaya Jawa di era digital ini. Adanya teknologi baru saat ini seperti Speech recognition memudahkan generasi milenial dalam mempelajari bahasa jawa atau bisa disebut juga dengan Automatic speech recognition (ASR) memproses input ucapan menjadi sinyal dengan mengonversi gelombang suara yang dapat dipahami komputer dan diklasifikasikan sebagai kata-kata atau memberikan balasan [1].

Hasil dari proses kata yang diucapkan ditampilkan dalam bentuk tulisan dan dapat dibaca oleh perangkat teknologi sebagai sebuah perintah. Konversi ucapan dari suara ke teks di aplikasi penerjemah ini yaitu sangat berguna bagi siswa tuna rungu dan orang-orang cacat fisik. Sinyal dapat stabil, yang berarti nilai statistiknya tidak berubah seiring berjalanannya waktu, atau tidak stabil, yang berarti nilai statistiknya berubah terhadap waktu. Pemodelan sinyal yang benar memungkinkan percobaan sumber dan pelatihan sebanyak mungkin selama proses simulasi. Singkatnya, ada tiga proses umum yang digunakan komputer untuk mengubah sinyal suara menjadi teks *preprocessing*, ekstraksi fitur, dan pengklasifikasi [2]. Jawa adalah salah satu bahasa daerah yang memiliki sejarah dan budaya yang kaya, dan masih digunakan oleh jutaan orang di berbagai daerah di Indonesia, terutama di pulau Jawa. Dijelaskan sebagai berikut:

1) Kebutuhan Komunikasi

Bahasa Jawa masih digunakan dalam percakapan sehari-hari di banyak daerah di Indonesia. Dengan adanya aplikasi penerjemah ini, orang

yang tidak fasih berbahasa Jawa tetapi perlu berkomunikasi dengan orang-orang yang menggunakan bahasa Jawa dapat melakukan komunikasi dengan lebih mudah.

2) Pelestarian Bahasa dan Budaya

Aplikasi ini juga dapat membantu dalam pelestarian bahasa dan budaya Jawa. Dengan memudahkan orang untuk belajar bahasa Jawa dan menggunakan bahasa tersebut dalam percakapan sehari-hari, aplikasi ini dapat membantu mencegah kepunahan bahasa Jawa.

3) Pendidikan

Aplikasi penerjemah juga dapat digunakan dalam konteks pendidikan. Siswa dan pelajar dapat menggunakan aplikasi ini untuk memahami teks atau materi pelajaran yang menggunakan bahasa Jawa. Ini dapat membantu meningkatkan pemahaman dan keterampilan berbahasa Jawa mereka.

4) Pengembangan Ekonomi

Aplikasi penerjemah Indonesia-Jawa juga dapat mendukung pertumbuhan ekonomi dengan memfasilitasi perdagangan dan bisnis antara komunitas berbicara Jawa dengan komunitas yang berbicara bahasa Indonesia. Dengan memahami bahasa satu sama lain, pelaku bisnis dapat menjalin kerjasama dan mengembangkan koneksi bisnis yang lebih baik.

Rumusan Masalah

Berdasarkan analisis dari latar belakang tersebut, maka dapat simpulkan rumusan masalah dari penelitian ini yaitu bagaimana pemanfaatan aplikasi penerjemah menggunakan *Automatic speech recognition* sebagai teknologi dalam meningkatkan minat generasi milenial untuk mempelajari bahasa jawa [3].

Speech Recognition

Pengenalan ucapan otomatis (ASR) adalah teknologi yang digunakan oleh perangkat lunak ketika mereka menerima input kata yang diucapkan. Perangkat menggunakan teknologi ini untuk mengenali dan memahami kata-kata yang diucapkan dengan mendigitalkannya dan mencocokkan sinyal digital dengan pola tertentu yang tersimpan di dalamnya. Setelah sekumpulan angka dibuat dari sekumpulan kata yang diucapkan, kode tertentu digunakan untuk membedakan antara angka-angka tersebut. Dalam sebuah buku tentang kecerdasan buatan (AI), pengenalan suara juga disebut sebagai proses

mengidentifikasi suara berdasarkan kata-kata yang diucapkan dengan mengubah sinyal peralatan audio. Banyak upaya yang diperlukan untuk menjadi suara bermanfaat, seperti pengenalan suara atau ucapan [4].

Flutter

Flutter adalah SDK seluler untuk *Flutter*, yang dirancang dan dikembangkan secara terbuka oleh Google, menyediakan solusi lengkap untuk pembuatan aplikasi, termasuk mesin rendering, komponen antarmuka pengguna (UI), kerangka kerja pengujian, alat pengembangan, sistem router, dan berbagai fitur lainnya [5]. *Flutter* juga merupakan SDK sumber terbuka yang didukung sepenuhnya oleh Google. Bahasa pemrograman C, C++, dan Dart digunakan untuk menulis *Flutter*, dan Skia Graphics Engine juga digunakan. Keberadaan berbagai widget yang dapat sepenuhnya disesuaikan, seperti perpustakaan Desain Material, adalah salah satu keunggulan utamanya.

Dart

Dart, seperti kebanyakan bahasa pemrograman lainnya, mungkin sulit untuk dipelajari. *Dart* memiliki keunggulan sebagai bahasa yang relatif "aman" untuk dipelajari, sehingga jika Anda akrab dengan JavaScript, Java, atau bahasa tingkat tinggi lainnya, Anda akan merasa familiar dengan sintaksis *Dart*. Desain berorientasi objek juga dapat menjadi akrab bagi mereka yang telah berpengalaman dengan Ruby atau Python. Sebaliknya, Google membuat *Dart* sebagai bahasa yang sederhana dan produktif yang dapat dikompilasi menjadi JavaScript. *Dart* telah ditunjukkan untuk berhasil dalam menulis antarmuka pengguna. *Dart* adalah bahasa pemrograman yang konservatif yang dimaksudkan untuk membuat lingkungan pemrograman yang stabil, konsisten, dan ramah pengembang [6].

Dart diciptakan oleh Google pada tahun 2011 dengan tujuan menggantikan JavaScript sebagai bahasa web. Fiturnya termasuk kemudahan penulisan kode, dukungan untuk pemrograman asinkron, dan kompilasi JIT (*Just-In-Time*) dan AOT (*Ahead-Of-Time*), yang memungkinkan pengembangan dan eksekusi kode yang cepat. Meskipun *Dart* menggunakan sintaksis yang akrab bagi pengembang yang sudah terbiasa dengan gaya bahasa C, beberapa elemen seperti loop, pernyataan if, dan pernyataan switch tetap sama dengan bahasa berbasis kelas

seperti Java, bahasa ini mencoba mengatasi kekurangan JavaScript dalam hal fleksibilitas dengan memperkenalkan sistem tipe yang lebih ketat.

Translator

Bahasa sebagai alat komunikasi manusia ditunjukkan melalui penerjemahan. Penerjemahan adalah bagian penting dari pertukaran informasi, gagasan, dan ilmu pengetahuan. Namun, konsekuensi yang tidak menguntungkan dapat muncul jika penerjemah tidak dapat menyampaikan arti dengan akurat sesuai dengan bahasa sumbernya. Oleh karena itu, memahami teks bahasa sumber dan gaya bahasa yang digunakan sangat penting bagi pembelajar penerjemahan. Saat ini, penerjemahan menggunakan metode konvensional dan alat penerjemah otomatis seperti CAT Tools [7].

Literasi digital sangat penting, dan orang harus mampu menggunakan media dan alat digital untuk memudahkan penerjemahan dan menghasilkan hasil yang baik. Singkatnya, dalam dunia penerjemahan kontemporer, memahami dengan baik bahasa sumber dan literasi digital sangat penting. Untuk memahami dan menghasilkan penerjemahan yang baik, penerjemahan memerlukan beberapa langkah atau siklus. Namun, tidak banyak orang yang menggunakan teknologi terbaru seperti penerjemahan mesin otak (NMT) atau mesin penerjemah berbasis AI. Namun, kemajuan teknologi kecerdasan buatan, terutama dalam bidang mesin penerjemah, telah menunjukkan bahwa itu efektif dalam meningkatkan keterampilan bahasa siswa. Akibatnya, penulis merencanakan untuk memberikan pelatihan penerjemahan dengan menggunakan *Google Translate* sebagai kecerdasan buatan dan penerapan strategi penerjemahan.

Google Cloud Platform

Google Cloud Platform (GCP) diluncurkan pada bulan November 2011 sebagai platform cloud publik yang menyediakan berbagai layanan komputasi, penyimpanan, jaringan, basis data, Big Data, keamanan, dan identitas. GCP berasal dari *Google App Engine* yang dirilis pada tahun 2008, dan sejak saat itu berkembang menjadi platform terintegrasi yang menyediakan layanan kontainer dan dukungan yang diperlukan untuk menjalankan beban kerja produksi dengan aman dan andal. GCP saat ini hadir di dua puluh negara dengan enam puluh satu pusat data, dan

menawarkan sekitar sembilan puluh produk dan layanan cloud yang mencakup berbagai kebutuhan bisnis. Melalui produk-produknya seperti *Google Cloud Platform* (GCP), *Google App Engine*, *Cloud Firestore*, dan *Firebase*. *Google Cloud Computing* menyediakan layanan *platform* sebagai layanan (*Platform as a Service/PaaS*) dan layanan *serverless*. *Google App Engine* adalah layanan *PaaS* yang memungkinkan pengguna mengunggah aplikasi web ke infrastruktur Google tanpa khawatir tentang spesifikasi server. Layanan penyimpanan data *Cloud Firestore* memudahkan penyinkronan aplikasi, membuat kueri data, dan menyimpan data. Sebagai *Backend as a Service* (BaaS), *Firebase* menawarkan berbagai alat dan layanan yang memungkinkan pengembangan aplikasi yang lebih cepat, seperti kemampuan untuk mengelola otentikasi pengguna. Perusahaan dapat menggunakan layanan ini untuk membuat aplikasi bisnis dengan mudah dan efisien.

Natural language processing

Natural language processing (NLP) adalah untuk membuat program khusus dalam bidang linguistik yang memungkinkan mesin untuk memahami bahasa manusia. Teks bahasa alami memiliki banyak tingkat entri bahasa dan banyak domain yang berbeda, dan merupakan data yang kompleks dan tidak terstruktur. Segmentasi kata, penandaan bagian ucapan, pengenalan entitas bernama, ekstraksi relasi, dan terjemahan mesin adalah beberapa tugas proses pengolahan bahasa natural. Pembelajaran representasi memungkinkan representasi otomatis dari data bahasa alami dalam berbagai tingkat dan domain. Dengan memfasilitasi transfer pengetahuan di antara berbagai tingkat bahasa, tugas, dan domain aplikasi, pembelajaran representasi memainkan peran penting dalam meningkatkan efektivitas dan ketahanan kinerja NLP. Aplikasi terkomputerisasi yang disebut *Natural Language Processing* (NLP) menggunakan teori dan teknik untuk menganalisis konten tekstual [8].

Proses pengolahan bahasa (NLP) memungkinkan orang bertukar kata-kata dalam berbagai bahasa, yang membantu mereka memahami dunia dengan memberikan wawasan yang akurat. Ini adalah sistem komputer yang menggabungkan teknik komputer untuk mempelajari, mengenali, dan meningkatkan pemahaman terhadap konten bahasa manusia. Proses pemrosesan bahasa alami (NLP) adalah cabang dari kecerdasan buatan yang bertujuan untuk

mengajarkan komputer dan perangkat digital untuk memahami bahasa manusia secara alami. Dalam konteks FinTech, pengolahan bahasa alami memainkan peran penting dalam mendukung teknologi "ABCDE", yang terdiri dari kecerdasan buatan, data besar, komputasi cloud, *blockchain*, dan *Internet of Things*. Dengan bantuan pemahaman bahasa alami, NLP membantu mengintegrasikan teknologi-teknologi ini ke dalam berbagai saluran *FinTech*.

Figma

Figma adalah alat desain UX/UI *real-time* yang kuat yang cocok untuk pengguna pemula dan berpengalaman. Dengan akses yang mudah melalui aplikasi desktop dan situs web, aplikasi ini dapat berfungsi sebagai pengganti aplikasi desain lainnya. Karena berbasis *cloud*, *Figma* dapat digunakan di berbagai sistem operasi, termasuk *Windows*, *Mac*, *Linux*, dan *ChromeOS*, tetapi belum resmi mendukung *iPadOS*. Dengan menyinkronkan perubahan secara online, *Figma* juga dapat menyimpan perubahan secara lokal jika koneksi internet hilang. desain grafis berbasis web.

Figma memiliki keunggulan dalam hal kemudahan penggunaan dan aksesibilitas karena dapat diakses melalui browser web tanpa mengunduh aplikasi terlebih dahulu. Berbagai proyek desain menggunakan *Figma*, mulai dari perancangan website hingga aplikasi mobile. Beberapa contohnya adalah desain UI/UX untuk website *Laboratorium Energy*, pengembangan UI/UX untuk aplikasi *Waste4Change*, dan desain aplikasi mobile untuk anak-anak dengan disleksia.

Struktur Navigasi

Struktur navigasi non-linier adalah tata letak yang tidak mengikuti urutan tertentu dan merupakan pengembangan dari struktur navigasi linier. Percabangan disarankan dalam struktur non-linier ini. Ketika orang membuka aplikasi, halaman utama pertama yang mereka lihat adalah tombol bantuan, yang terletak di pojok kanan atas halaman. Selain itu, ada menu bantuan yang berisi informasi tentang situs web ini.

Menu ini mencakup daftar perintah yang dapat diakses, instruksi penggunaan web, dan jumlah objek yang dapat ditemukan rancangan program yang digunakan untuk membuat program lebih mudah digunakan dan memberikan gambaran umum kepada

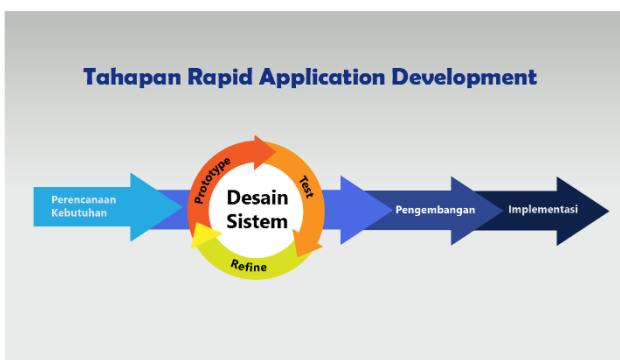
pengguna. Memanfaatkan data sistem dan instruksi yang terdapat dalam pilihan menu, struktur navigasi memungkinkan pengguna merancang dan menjalankan aplikasi [9].

Mobile Application

Android adalah istilah untuk kumpulan perangkat lunak perangkat bergerak yang berbeda yang terdiri dari aplikasi, *middleware*, dan sistem operasi. Android menyediakan sejumlah alat yang dapat diakses langsung oleh pengembang, yang memungkinkan pengembang untuk menggunakan *library* dan *tools* untuk membangun aplikasi yang lebih baik. sehingga hal ini dapat membantu pengembang meningkatkan efisiensi saat mereka membuat aplikasi. Selain SMS dan panggilan telepon, konsumsi media digital, terutama melalui ponsel dan aplikasi, telah menjadi kenyataan yang jelas sejak akhir-akhir ini.

Pengembangan Aplikasi Seluler tidak semudah yang terlihat. Kita telah melihat berbagai jenis aplikasi yang dapat dikembangkan untuk pengembangan aplikasi. Sekarang kita akan melihat siklus hidup pengembangan aplikasi, terlepas dari kompleksitas aplikasi yang sedang dikembangkan. Tujuan dan Persyaratan Aplikasi Seluler Menetapkan tujuan pembuatan aplikasi adalah bagian penting dari pengembangan. Ini menjadi tujuan pelanggan untuk menentukan tujuan apa yang harus dicapai oleh aplikasi [10].

2. Metode Penelitian



Gambar 1. Metode Penelitian

Metode penelitian ini menggunakan algoritma *Natural Language Processing* (NLP) ada beberapa pendekatan untuk menerjemahkan teks. Salah satu pendekatan adalah menggunakan model terjemahan

statistik atau model terjemahan neural. NMT adalah jenis algoritma penerjemahan mesin yang menggunakan jaringan saraf untuk memahami konteks dan membuat terjemahan yang lebih akurat dan alami. Model-model ini telah melibatkan pembelajaran mesin yang mendalam dan besar dataset terjemahan untuk memahami korelasi antara kata-kata dalam dua bahasa. Pembangunan Model Bahasa (*Language Model*) adalah proses yang melibatkan representasi statistik dari bahasa. Dalam konteks terjemahan, model bahasa memahami struktur kalimat, kosakata, dan sintaksis.

Proses terjemahan dimulai setelah teks diproses dan model bahasa dibangun. Model menerima teks dalam satu bahasa dan menghasilkan terjemahan dalam bahasa lain. Hasil terjemahan dievaluasi untuk akurasi, dan model dapat disesuaikan dengan *fine-tuning* atau perubahan arsitektur jika terdapat kesalahan. Setelah evaluasi, hasil terjemahan mungkin perlu melalui tahap *post-processing* untuk memastikan struktur kalimat yang benar. Terjemahan dapat diintegrasikan ke dalam aplikasi pengguna melalui antarmuka grafis atau antarmuka pemrograman aplikasi (API).

Waktu Penelitian

Waktu yang dibutuhkan untuk menyelesaikan proses penelitian dan pengembangan aplikasi adalah 19 minggu atau kurang lebih 5 bulan. Berikut di bawah ini adalah tabel timeline dari awal hingga akhir.

Tabel 1. Timeline

No	Dilakukan	Bulan				
		Oktober		November	Desember	Januari
		1 to 30	1 to 31	1 to 30	1 to 31	1 to 30
1	Pendefinisi masalah					
2	Desain					
4	Pengembangan					
5	Uji coba					
6	Publikasi					

Penentuan Subjek Penelitian

Sistem aplikasi penerjemah ini dibangun dari beberapa sumber seperti refensi jurnal penelitian, serta algoritma yang digunakan untuk membangun fitur yang ada dalam aplikasi ini. Penggunaannya aplikasi ini ditujukan untuk generasi milenial yang ingin mempelajari bahasa jawa untuk berkomunikasi. Diharapkan dengan dibangunnya aplikasi penerjemah ini bisa mempermudah dalam mempelajarinya.

Fokus Penelitian

Fokus penelitian ini adalah melakukan pembuatan aplikasi penerjemah, Penerapan fitur speech to text menjadi fokus utama dalam penelitian ini, dan implementasi algoritma natural language processing terkait proses penerjemah ucapan ke kata yang diterjemahkan. Penerapan algoritma Sequential terkait pencarian data mahasiswa dan dosen.

Sumber Data

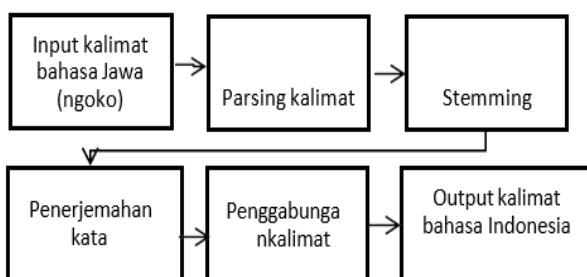
Tahap penelitian saat ini sedang dilakukan, dan data dikumpulkan dari artikel terkait yang ditulis oleh peneliti sebelumnya untuk digunakan sebagai sumber referensinya. Nantinya sumber ini digunakan untuk membantu membuat aplikasi.

Teknik Pengumpulan Data

Data yang digunakan pada aplikasi penerjemah ini dari berbagai artikel artikel untuk mencakup informasi yang sudah ada.

Struktur Desain Aplikasi

Aplikasi yang akan dibangun ini merupakan penerjemah otomatis bahasa Jawa ke dalam bahasa Indonesia dengan metode *rule-based* berdasarkan analisis morfologi. Aplikasi ini berbasis desktop, sehingga harus dilakukan penginstallan terlebih dahulu untuk dapat menggunakan kapanpun dan dimanapun. Sistem ini dirancang dan dibangun secara bertahap. Tahapan awal dari pembuatan aplikasi ini yaitu penginputan kalimat bahasa Jawa, parsing kalimat yang terdiri dari proses case folding dan tokenizing, terjemahkan kata, menganalisa imbuhan dengan aturan analisis kontrastif, lalu yang terakhir penggabungan kata menjadi kalimat bahasa Indonesia. Secara umum dari proses pembuatan aplikasi penerjemah otomatis ini akan digambarkan sebagai berikut.



Gambar 2. Proses Pembuatan Aplikasi Penerjemah

Tahapan-tahapan umum yang dilakukan dalam membangun aplikasi penerjemah otomatis bahasa Jawa ke dalam bahasa Indonesia adalah penginputan kalimat bahasa Jawa (ngoko) oleh *user*, lalu dilanjutkan dengan proses *case folding* yaitu semua huruf pada kalimat yang diinputkan oleh user akan diubah menjadi huruf kecil. Setelah proses *case folding* dilanjutkan dengan proses *tokenizing*. Proses *tokenizing* ini melakukan pemecahan kalimat dengan menghilangkan tanda baca maupun simbol-simbol lain. Pemecahan kalimat yang dilakukan berdasarkan spasi sehingga diperoleh potongan kata. Kemudian, dilakukan *stemming* yaitu pemotongan imbuhan dari kata menjadi kata dasar yang sesuai dengan morfologi bahasa Jawa yang baik dan benar.

Proses *case folding*, *tokenizing*, dan *stemming* yang sudah dilakukan pada kata dasar, dilanjutkan dengan perhitungan *cosine similarity* guna menemukan kandidat kata yang mirip dengan kata dasar yang dimaksud dalam kalimat lalu diterjemahkan ke dalam bahasa Indonesia dan digabungkan kata-kata tersebut menjadi suatu kalimat yang merupakan output dari kalimat yang telah diterjemahkan yaitu bahasa Jawa ke dalam bahasa Indonesia.

Desain Proses

Input Kalimat Bahasa Jawa

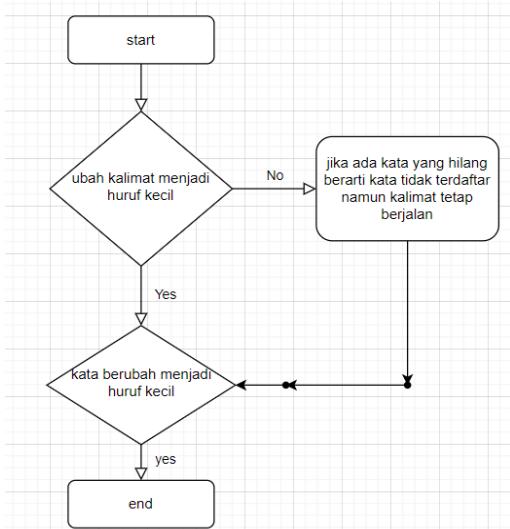
Kalimat yang diinputkan dalam sistem ini yaitu kalimat bahasa Jawa. Dari tiap kalimat akan diproses lebih lanjut untuk memberikan proses yang maksimal dengan melakukan parsing di dalamnya agar menghasilkan terjemahan yang sesuai dengan kalimat yang diinputkan

Parsing

Pada tahap parsing dalam penelitian ini meliputi *case folding* dan *tokenizing*. Tahapan yang dilakukan sebelum proses pemotongan kata berimbuhan yang akan menghasilkan sebuah kata dasar dari bentuk kata asli.

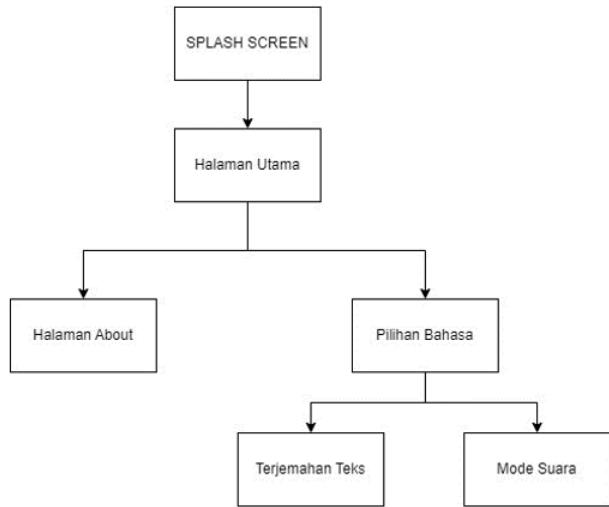
Case Folding

Proses awal dari rangkaian *preprocessing* teks atau dokumen adalah *case folding*. Dalam proses *case folding* ini akan dilakukan pengubahan seluruh karakter yang merupakan huruf kapital menjadi huruf kecil (hanya huruf a sampai dengan z yang diterima). Karakter selain huruf dihilangkan dan dianggap sebagai delimiter. Berikut alur proses tahapan dari *case folding*.



Gambar 3. Alur Proses Case Folding

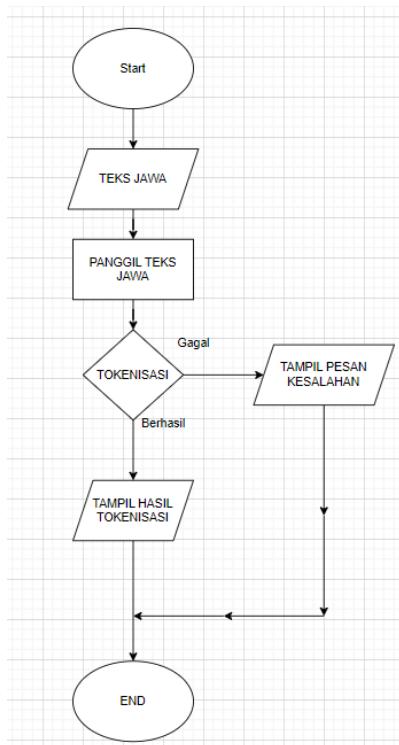
Struktur aplikasi



Gambar 5. Struktur Aplikasi

Tokenizing

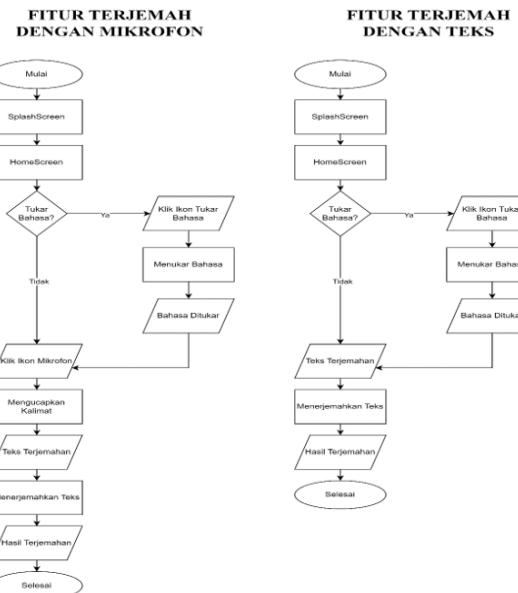
Setelah proses perubahan ukuran huruf pada kalimat, dilanjutkan dengan proses memisahkan kata yang ada pada dokumen berdasarkan spasi kemudian memproses kata yang telah dipisahkan tersebut ke dalam sebuah tabel untuk dilakukan proses berikutnya. Dalam hal ini dinamakan *tokenizing*. *Tokenizing* yaitu proses pemisahan suatu karakter berdasarkan karakter spasi.



Gambar 4. Tokenizing

Pada gambar struktur diatas terdapat tata cara sebagai berikut. Aplikasi dimulai dari *Splash Screen* sebagai halaman awal. Setelah *Splash Screen*, pengguna secara default menuju ke Halaman *Home*, yang memiliki fitur utama berikut:

- 1) Mikrofon untuk terjemahan suara.
- 2) *Text field* untuk mengisi teks yang ingin diterjemahkan.
- 3) *Switch Bahasa* untuk menukar bahasa yang akan diterjemahkan.



Gambar 6. Struktur Fitur-Fitur

Pada gambar struktur fitur fitur diatas terdapat penjelasan sebagai berikut.

- 1) Pengguna dapat memilih mode terjemahan, yaitu "Mode Suara" atau "Mode Teks." Mode Suara memungkinkan pengguna untuk berbicara dan menerjemahkan suara ke bahasa Jawa. Mode Teks memungkinkan pengguna untuk memasukkan teks yang ingin diterjemahkan.
- 2) Jika pengguna memilih "Mode Suara," aplikasi akan mengaktifkan mikrofon dan mendengarkan suara pengguna. Suara tersebut akan diterjemahkan ke bahasa Jawa.
- 3) Jika pengguna memilih "Mode Teks," mereka dapat memasukkan teks dalam bahasa Indonesia ke dalam *text field*. Tekstual akan diterjemahkan ke bahasa Jawa.
- 4) Pengguna dapat menggunakan *Switch* Bahasa untuk menukar bahasa yang akan diterjemahkan, baik dari bahasa Indonesia ke bahasa Jawa atau sebaliknya.
- 5) Setelah pengaturan bahasa selesai, aplikasi akan memproses teks atau suara yang dimasukkan dan menghasilkan terjemahan.
- 6) Hasil terjemahan akan ditampilkan di layar.
- 7) Aplikasi selesai, dan pengguna dapat melakukan tindakan lain sesuai kebutuhan.

3. Hasil dan Pembahasan

Hasil Penelitian

Dalam penelitian ini, dilakukan analisis kinerja aplikasi penerjemah dari bahasa Indonesia ke bahasa Jawa. Beberapa metrik yang diukur melibatkan kecepatan terjemahan, akurasi, dan kemampuan aplikasi dalam menangani frasa atau kosakata khusus bahasa Jawa. Hasil pengujian menunjukkan bahwa aplikasi ini memiliki kecepatan yang cukup baik dalam menerjemahkan kalimat sederhana, namun mengalami tantangan ketika menghadapi frasa atau kata-kata khas bahasa Jawa yang kurang umum. Selanjutnya, hasil terjemahan dievaluasi dengan membandingkan terjemahan aplikasi dengan terjemahan manusia yang dilakukan oleh penerjemah berbahasa Jawa yang berpengalaman. Evaluasi ini bertujuan untuk mengukur tingkat akurasi dan keberterimaan hasil terjemahan. Meskipun aplikasi mampu memberikan terjemahan yang cukup memadai untuk kalimat umum, terdapat beberapa

kesalahan dalam menangani struktur kalimat yang kompleks dan istilah khusus. Berdasarkan hasil penelitian ini, terdapat beberapa implikasi yang dapat diidentifikasi.

Pertama, perlu dilakukan perbaikan pada kemampuan aplikasi dalam mengatasi struktur kalimat yang kompleks dan kosakata khusus bahasa Jawa. Kedua, pengembang dapat mempertimbangkan peningkatan algoritma untuk meningkatkan akurasi terjemahan. Secara keseluruhan, hasil penelitian ini memberikan pemahaman yang lebih mendalam tentang kinerja aplikasi penerjemah dari bahasa Indonesia ke bahasa Jawa. Meskipun masih ada beberapa tantangan yang perlu diatasi, potensi aplikasi ini untuk meningkatkan komunikasi antarbahasa tetap signifikan. Dengan implementasi perbaikan yang tepat, diharapkan aplikasi ini dapat menjadi alat yang lebih efektif dan dapat diterima oleh pengguna berbahasa Jawa.

Tahap Analisis

Pada tahap analisis dilakukan dengan mengumpulkan informasi yang dibutuhkan untuk pembuatan aplikasi. Tahap analisis meliputi analisis fungsional dan nonfungsional.

Analisis Fungsional

Analisis fungsional merupakan analisis terhadap fungsi-fungsi yang diperlukan pada aplikasi penerjemah Jawa antara lain:

- 1) Aplikasi dapat melakukan proses pencarian terjemahan menggunakan suara.
- 2) Aplikasi dapat menghasilkan suara yang diterjemahkan.
- 3) Aplikasi dapat melakukan pertukaran bahasa asal dan tujuan
- 4) terjemahan.
- 5) Aplikasi dapat melakukan salin dan share ke media sosial arti penerjemahan.

Analisis Nonfungsional

Analisis nonfungsional merupakan analisis kebutuhan yang diperlukan selama pengembangan aplikasi. Analisis nonfungsional meliputi analisis isi program, analisis *hardware*, dan analisis *software*.

1) Analisis Isi Program

Tahap analisis isi program terkait materi yang akan disajikan. Konten materi yang disajikan bersumber *Google Cloud Platform*.

2) Analisis Hardware

Analisis hardware meliputi spesifikasi *hardware* yang mendukung untuk pengembangan aplikasi.

Dukungan *hardware* yang dibutuhkan yaitu hardware yang dapat menjalankan lingkungan pengembangan Android Studio serta Visual Studio Code. Berikut spesifikasi hardware minimal untuk Android Studio.

- a) Microsoft Windows 7/8/10
- b) 3 GB RAM
- c) 2 GB hard disk space

Sedangkan untuk spesifikasi minimal Visual Studio Code adalah sebagai berikut.

- a) Microsoft Windows 7/8/10
- b) Intel Core i3 atau AMD Athlon 64
- c) 2 GB RAM
- d) 2 GB hard disk space

Pengembangan aplikasi Ukara Jawa menggunakan spesifikasi yang disarankan yaitu:

- a) Microsoft Windows 7
- b) Intel Core i5
- c) 4 GB RAM

Untuk *hardware* yang dapat menjalankan aplikasi Ukara Jawa yaitu smartphone yang memiliki sistem operasi Android dengan versi 4.0.x atau yang terbaru.

Desain Aplikasi

Pada tahap ini dilakukan pembuatan desain aplikasi penerjemah secara detail yang digunakan sebagai acuan dalam pembuatan aplikasi berbahasa daerah yang baru nantinya, sehingga akan menjelaskan rangkaian dari aplikasi penerjemah yang nantinya akan diterapkan. Selain itu pembuatan desain aplikasi penerjemah juga dapat dipergunakan sebagai acuan untuk menarik perhatian yang ingin belajar bahasa.

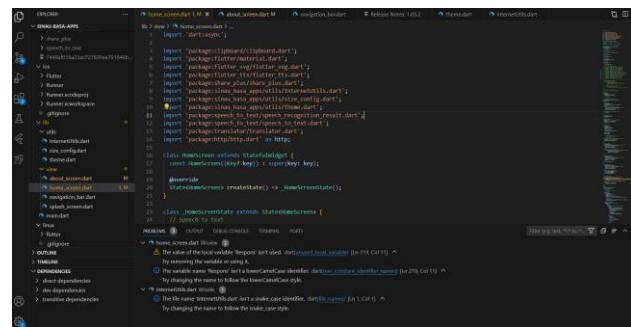
Tahap Implementasi

Pada tahap implementasi aplikasi sinaubasa ini menggunakan Visual Studio Code dengan target pengembangan aplikasi mulai dari versi 4.0 sampai dengan versi 7.1. Tahapan implementasinya adalah sebagai berikut.

Implementasi Layout Interface Aplikasi

Implementasi *layout interface* aplikasi menggunakan script XML yang merupakan standar pengaturan

layout Android. Layout yang digunakan pada aplikasi antara lain *about screen*, *home screen*, *splash screen*.



Gambar 7. Implementasi *layout interface*

Berikut gambar tampilannya:



Gambar 8. *Splash Screen*

Splash screen atau yang bisa disebut juga halaman awal dari aplikasi. Aplikasi ini bernama sinau basa dalam bahasa jawa yang artinya belajar bahasa.



Gambar 9. *About*

About yaitu halaman penjelasan tentang aplikasi yang dibuatnya halaman ini membahas tentang profil pembuat dan penjelasan singkat apa yang dibuatnya.



Gambar 10. Tampilan Home

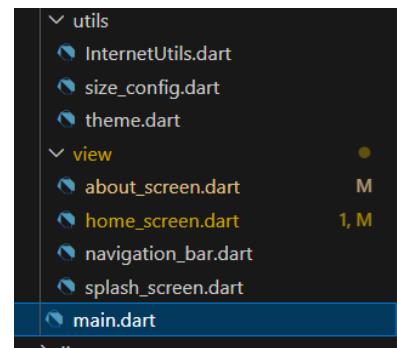
Halaman *Home* yaitu halaman yang menyediakan fitur fitur penerjemah mulai dari teks maupun suara. Aplikasi penerjemah ini juga mempunyai fitur tukar bahasa jadi tidak dari bahasa indonesia saja yang bisa terjemah ke bahasa jawa tapi bahasa jawa juga bisa menerjemahkan ke bahasa indonesia dan bukan itu saja aplikasi ini juga bisa mengeluarkan suara yang kita tulis. Aplikasi ini juga menyediakan menyalin text dan share untuk memudahkan pengguna untuk memakainya.

Implementasi Logika Pemrograman

Aplikasi tidak dapat dijalankan apabila tidak ada perintah yang digunakan. Perintah yang dimaksudkan untuk menjalankan fungsi-fungsi pada aplikasi. Untuk itu diperlukan bahasa pemrograman supaya fungsi dapat dijalankan. Android secara default menggunakan bahasa pemrograman dart. Pada aplikasi sinaubasa file yang digunakan dibagi ke dalam empat *package* yang meliputi package utils, view dan main.

Tabel 2. Hasil pengujian *functionality suitability*

No	Aktivitas/Menu	Skenario Uji	Pertanyaan	Ketercapaian	
				Ya	Tidak
1	Membuka Aplikasi	Memilih ikon launcher aplikasi Sinaubasa	Apakah halaman utama dapat ditampilkan dan muncul semua menu (Terjemahan, Bantuan, Info Pengembang, dan Keluar) dengan benar ?	3	
2	Halaman Utama	Memilih menu Terjemahan	Apakah pengguna dapat membuka menu Terjemahan ?	3	
		Menu Terjemahan (menekan tombol	Apakah pengguna dapat kembali ke halaman utama dengan menekan tombol	3	



Gambar 11. Implementasi Logika Pemrograman

```

53 // List<Map<String, String>> kamustata = [
54   {"Indonesia": "selamat pagi", "jawa": "sugeng ening"}, 
55   {"Indonesia": "selamat siang", "jawa": "sugeng siyang"}, 
56   {"Indonesia": "selamat sore", "jawa": "sugeng dalu"}, 
57   {"Indonesia": "selamat malam", "jawa": "sugeng dalu"}, 
58   {"Indonesia": "terima kasih", "jawa": "matur manuh"}, 
59   {"Indonesia": "saya suka makan", "jawa": "makan seud"}, 
60   {"Indonesia": "saya suka main", "jawa": "main kuhure"}, 
61   {"Indonesia": "saya suka cuci", "jawa": "menengkor"}, 
62   {"Indonesia": "saya bisa", "jawa": "sampaian sagep munung"}, 
63   {"Indonesia": "selamat makan", "jawa": "senggo dininkati suguhane"}, 
64   {"Indonesia": "jangan lupa", "jawa": "ojo lali"}, 
65   {"Indonesia": "di mana kana sandi", "jawa": "endi jedhing"}, 
66   {"Indonesia": "aku menyukaimu", "jawa": "aku seneng karo kowe"}, 
67   {"Indonesia": "jangan berceandu", "jawa": "ojo guyon"}, 
68   {"Indonesia": "jangan berceandu", "jawa": "jangan ngapusi"}, 
69   {"Indonesia": "saya hanya", "jawa": "aku ngelak"}, 
70   {"Indonesia": "saya lapar", "jawa": "aku lueh"}, ...

```

Gambar 12. Implementasi Algoritma Natural Language Processing dari Bahasa Indonesia ke Bahasa Jawa

Tahap Pengujian

Tahap terakhir dari penelitian ini adalah dengan melakukan pengujian. Pengujian yang dilakukan menggunakan instrumen penelitian sesuai dengan standar ISO 25010 yaitu *functionality suitability, compatibility, usability, reliability*, serta uji materi. Hasil pengujian adalah sebagai berikut.

Pengujian Functionality Suitability

Pengujian *functionality suitability* dilakukan menggunakan kuesioner checklist yang diisi oleh ahli software sebanyak tiga orang. Hasil pengujian *functionality suitability* adalah sebagai berikut.

back)	back ?	
Memilih menu Bantuan	Apakah halaman Bantuan Penggunaan dapat ditampilkan dengan benar ?	3
Memilih menu Info Pengembang	Apakah halaman Info Pengembang dapat ditampilkan dengan benar ?	3

Pengujian compatibility

Pengujian *compatibility* dilakukan dengan pengujian aspek *co-existence*, pengujian berbagai sistem operasi dan jenis perangkat. Berikut ini ringkasan hasil pengujian *compatibility*.

Pengujian aspek co-existence

Pengujian aspek *co-existence* dilakukan dengan menjalankan aplikasi Ukara Jawa bersamaan dengan aplikasi lain pada satu perangkat. Berikut ini hasil pengujian aspek *co-existence*.

Tabel 3. Pengujian aspek *co-existence*

No	Aplikasi yang dijalankan	Ketercapaian Hasil	
		Sukses	Gagal
1	Apakah aplikasi Sinaubasa dapat berjalan bersamaan dengan Whatsapp?	Ya	
2	Apakah aplikasi Sinaubasa dapat berjalan bersamaan dengan Line?	Ya	
3	Apakah aplikasi Sinaubasa dapat berjalan bersamaan dengan Google Translate?	Ya	
4	Apakah aplikasi Sinaubasa dapat berjalan bersamaan dengan Opera Mini?	Ya	
5	Apakah aplikasi Sinaubasa dapat berjalan bersamaan dengan Messaging?	Ya	
6	Apakah aplikasi Sinaubasa dapat berjalan bersamaan dengan Gmail?	Ya	
7	Apakah aplikasi Sinaubasa dapat berjalan bersamaan dengan Play Store?	Ya	
8	Apakah aplikasi Sinaubasa dapat berjalan bersamaan dengan Facebook?	Ya	

Pembahasan

Aplikasi Sinaubasa dikembangkan melalui tahap analisis, desain, implementasi, dan pengujian. Pada tahap analisis meliputi analisis fungisional dan analisis nonfungisional. Analisis fungisional dilakukan untuk mengumpulkan informasi terkait fungsi-fungsi yang dibutuhkan pada aplikasi Sinaubasa. Analisis nonfungisional digunakan untuk mengumpulkan informasi yang dibutuhkan dalam tahap pengembangan aplikasi Sinaubasa. Pada tahap desain terdiri atas *desain unified modelling language* (UML), desain sistem, dan desain interface. Desain UML digunakan untuk menggambarkan rancangan alur proses aplikasi terhadap sistem. Desain sistem digunakan untuk menggambarkan rancangan basis data serta alur kerja sistem yang digambarkan dengan *flowchart*.

Pembuatan *flowchart* alur kerja sistem menggunakan software Microsoft Visio 2013. Desain *interface* digunakan untuk menggambarkan tampilan *interface* aplikasi. Pembuatan desain interface dengan menggunakan *software* Corel Draw X7. Pada tahap implementasi terdiri atas implementasi *layout interface* aplikasi dan implementasi logika pemrograman.

Pembuatan *layout interface* aplikasi dan logika pemrograman menggunakan *software* visual studio code.

4. Kesimpulan

- 1) Pentingnya NLP dalam Penerjemahan Bahasa Algoritma NLP memiliki peran krusial dalam meningkatkan kualitas penerjemahan antara bahasa Indonesia dan Jawa. Kemampuan algoritma untuk memahami dan menerjemahkan struktur kalimat yang tepat menjadi kunci utama keberhasilan aplikasi ini.
- 2) Manfaat Metode *Speech Processing*: Penggunaan metode *Speech Processing* membuka pintu untuk interaksi suara yang lebih alami antara pengguna dan aplikasi. Hal ini dapat meningkatkan pengalaman pengguna dan membuat proses penerjemahan lebih cepat dan efisien.
- 3) Tantangan dalam Pengembangan Aplikasi. Meskipun potensi besar, pengembangan aplikasi penerjemah Indonesia-Jawa menggunakan algoritma NLP dan *Speech Processing* juga

menghadapi beberapa tantangan. Beberapa di antaranya termasuk kompleksitas bahasa, variasi dialek, dan kebutuhan akan set data yang besar untuk pelatihan yang efektif.

- 4) Upaya Penelitian dan Pengembangan Masa Depan. Kesimpulan makalah ini juga dapat menggarisbawahi pentingnya penelitian dan pengembangan lebih lanjut di bidang ini. Dengan terus memperbaiki algoritma dan mengatasi kendala teknis, kita dapat menghadirkan aplikasi penerjemah yang lebih canggih dan efektif.
- 5) Implikasi Sosial dan Budaya. Implementasi aplikasi ini juga memiliki implikasi sosial dan budaya yang signifikan. Penerjemahan antar-bahasa dan dialek dapat memperkuat komunikasi lintas budaya dan mendukung pelestarian bahasa lokal.

Untuk meningkatkan kemampuan aplikasi dalam mengatasi beragam konteks linguistik, disarankan untuk terus mengumpulkan data yang lebih banyak dan representatif dari variasi bahasa Indonesia dan Jawa, termasuk berbagai dialek. Data yang lebih luas dan mencakup variasi yang lebih besar akan memperkuat aplikasi. Selain itu, penelitian lebih lanjut diperlukan untuk mengoptimalkan algoritma NLP agar dapat menangani sintaksis dan semantik bahasa Indonesia dan Jawa dengan lebih baik, sehingga terjemahan menjadi lebih akurat dan kontekstual.

Integrasi teknologi AI yang lebih canggih, seperti machine learning dan deep learning, juga penting untuk meningkatkan kemampuan adaptasi algoritma terhadap perubahan bahasa dan dialek seiring waktu, yang akan meningkatkan fleksibilitas dan daya tanggap aplikasi. Fokus pada pengembangan antarmuka pengguna yang lebih intuitif, terutama dalam konteks interaksi suara, akan mempermudah penggunaan dan meningkatkan adopsi aplikasi oleh pengguna yang mungkin tidak terbiasa dengan teknologi NLP atau Speech Processing. Terakhir, melakukan uji coba terhadap komunitas pengguna potensial, khususnya mereka yang aktif menggunakan bahasa Indonesia dan Jawa dalam kehidupan sehari-hari, sangat penting karena feedback langsung dari pengguna dapat memberikan wawasan berharga untuk peningkatan lebih lanjut.

5. Daftar Pustaka

- [1] Muhammad, M., Hidayat, S., & Amrullah, A. Z. (2019). Speech Recognition Untuk Aplikasi Kamus Bahasa Indonesia-Sumbawa Berbasis Android. *Jurnal Bumigora Information Technology (BITe)*, 1(2), 126-137. DOI: <https://doi.org/10.30812/bite.v1i2.606>.
- [2] Adnan, F., & Amelia, I. (2022). Implementasi Voice Recognition Berbasis Machine Learning. *Edu Elektrika Journal*, 11(1), 24-29. DOI <https://doi.org/10.15294/eej.v11i1.57283>.
- [3] Asri, N. A. L., Adam, R. I., & Dermawan, B. A. (2023). SPEECH RECOGNITION UNTUK KLASIFIKASI PENGUCAPAN NAMA HEWAN DALAM BAHASA SUNDA MENGGUNAKAN METODE LONG-SHORT TERM MEMORY. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(2), 1242-1247. DOI: <https://doi.org/10.36040/jati.v7i2.6744>.
- [4] AL-Bakhrani, A. A., Amran, G. A., Al-Hejri, A. M., Chavan, S. R., Manza, R., & Nimbhore, S. (2023, August). Development of Multilingual Speech Recognition and Translation Technologies for Communication and Interaction. In *First International Conference on Advances in Computer Vision and Artificial Intelligence Technologies (ACVAIT 2022)* (pp. 711-723). Atlantis Press.
- [5] Alessandria, S. (2023). *Flutter Cookbook: 100+ step-by-step recipes for building cross-platform, professional-grade apps with Flutter 3.10. x and Dart 3. x*. Packt Publishing Ltd.
- [6] Arnesia, P. D., Pratama, N. A., & Sjafrina, F. (2022). Aplikasi Artificial Intelligence Untuk Mendeteksi Objek Berbasis Web Menggunakan Library Tensorflow Js, React Js Dan Coco Dataset. *JSII (Jurnal Sistem Informasi)*, 9(1), 62-69. DOI: <https://doi.org/10.30656/jsii.v9i1.4243>.
- [7] Bano, S., Jithendra, P., Niharika, G. L., & Sikhi, Y. (2020, November). Speech to text translation enabling multilingualism. In *2020 IEEE International Conference for Innovation in Technology*

- (INOCON) (pp. 1-4). IEEE. DOI: 10.1109/INOCON50539.2020.9298280.
- [8] Thomas, C. G., & Devi, J. (2021). A study and overview of the mobile app development industry. *International Journal of Applied Engineering and Management Letters (IJAEML)*, 5(1), 115-130.
- [9] Buana, I. K. S. (2020). Implementasi aplikasi speech to text untuk memudahkan wartawan mencatat wawancara dengan python. *Jurnal Sistem Dan Informatika (JSI)*, 14(2), 135-142. DOI: <https://doi.org/10.30864/jsi.v14i2.293>.
- [10] Ifrah, S. (2021). Getting Started with Containers in Google Cloud Platform. *Melbourne: VIC*.
- [11] Ilham, M., Kharisma, A. P., & Wardhono, W. S. (2019). Pengembangan Aplikasi Pembelajaran Membaca Aksara Jawa Menggunakan Speech Recognition (Studi Kasus: SDN Tulusrejo 1 Malang). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 3(9), 8640-8646.
- [12] Koalu, O., & Sompie, S. R. (2019). Rancang bangun aplikasi pengenalan bahasa tountemboan menggunakan speech recognition. *Jurnal Teknik Informatika*, 14(2), 269-278. DOI: <https://doi.org/10.35793/jti.14.2.2019.24003>.
- [13] Setiyanto, K., & Al Ghifari, H. R. (2023). APLIKASI PENJUALAN WAGHE COFFEE BERBASIS WEBSITE MENGGUNAKAN PHP AND MY SQL DENGAN FRAMEWORK BOOTSTRAP. *Jurnal Ilmiah Teknik*, 2(2), 51-61. DOI: <https://doi.org/10.56127/juit.v2i2.784>.
- [14] Lestari, A., & Hasibuan, A. (2022). Pergeseran Bahasa Jawa Pada Kalangan Remaja Desa Firdaus Kabupaten Serdang Bedagai. *Jurnal Pusat Studi Pendidikan Rakyat*, 139-147.
- [15] Liu, Y., Zhang, J., Xiong, H., Zhou, L., He, Z., Wu, H., ... & Zong, C. (2020, April). Synchronous speech recognition and speech-to-text translation with interactive decoding. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 34, No. 05, pp. 8417-8424). DOI: <https://doi.org/10.1609/aaai.v34i05.6360>.
- [16] Liu, Z., Lin, Y., & Sun, M. (2023). *Representation learning for natural language processing* (p. 521). Springer Nature.
- [17] Anissa, D. L. F., & Andryani, R. (2022). Penerapan Cloud Computing Dalam Aplikasi Panggil Teknisi Berbasis Android Menggunakan Google Cloud Platform. *J-SAKTI (Jurnal Sains Komputer dan Informatika)*, 6(2), 1292-1300. DOI: <http://dx.doi.org/10.30645/j-sakti.v6i2.533>.
- [18] Muliani, A. (2023). Aplikasi Kamus Bahasa Batak-Inggris (Speech To Speech) Berbasis Android Sebagai Fitur Pendukung Pariwisata Danau Toba Sebagai Destinasi Wisata Internasional. *JurTI (Jurnal Teknologi Informasi)*, 7(1), 145-151.
- [19] Sofi, N., & Dharmawan, R. (2022). Perancangan aplikasi bengkel CSM berbasis Android menggunakan framework Flutter (bahasa Dart). *Jurnal Teknik dan Science*, 1(2), 53-64. DOI: <https://doi.org/10.56127/jts.v1i2.125>.
- [20] Nugroho, K. (2019). Implementasi Sistem Speech To Text Berbasis Android Menggunakan APP Inventor Speech Recognizer. *Jurnal Ilmiah Infokam*, 15(1). DOI: <https://doi.org/10.53845/infokam.v15i1.167>.
- [21] Mah, P. M., Skalna, I., Muzam, J., & Song, L. (2022). Analysis of natural language processing in the fintech models of mid-21st Century. *Journal of Information Technology and Digital World*, 4(3), 183-211.
- [22] Pranaja, M. A., & Nurhidayat, A. I. PEMBUATAN APLIKASI MESIN PENERJEMAH MENGGUNAKAN METODE NO LANGUAGE LEFT BEHIND DARI BAHASA INDONESIA KE

BAHASA BANJAR.

- [23] Bangun, B. P. (2022). PENERAPAN NATURAL LANGUAGE PROCESSING UNTUK MEMBANGUN VOICE TRANSLATOR DENGAN BANTUAN NEURAL MACHINE TRANSLATION. *Deli Sains Informatika*, 1(2).
- [24] Putra, M. M. I., Sherwin, S., & Paturusi, S. (2020). Implementation of Speech Recognition in English Learning Applications for Children. *Jurnal Teknik Informatika*, 15(4), 247.
- [25] Surianto, D. F., Wahid, M. S. N., Parenreng, J. M., Wahid, A., Zain, S. G., Edy, M. R., & Risal, A. A. N. (2023). PKM Pelatihan Figma untuk Desain Prototipe Sistem Informasi. *Vokatek: Jurnal Pengabdian Masyarakat*, 57-63. DOI: <https://doi.org/10.61255/vokatekjpm.v1i2.88>.
- [26] Wahyuni, S., Layuk, N. S., Loly, R. H., & Daud, A. N. (2021). Desain sistem speech recognition penerjemah bahasa toraja menggunakan hidden markov model. *Jurnal Penelitian Pos dan Informatika*, 11(2), 107-119. DOI: <https://doi.org/10.17933/jppi.v11i2.286>.
- [27] Kamal, A. (2022). Pelatihan penerjemahan Indonesian-English dengan menggunakan kombinasi Google Translate dan menerapkan manual translation strategies. *Ash-Shababah: Jurnal Pengabdian Masyarakat*, 1(2), 9-15.